
# 8086 Flag Register

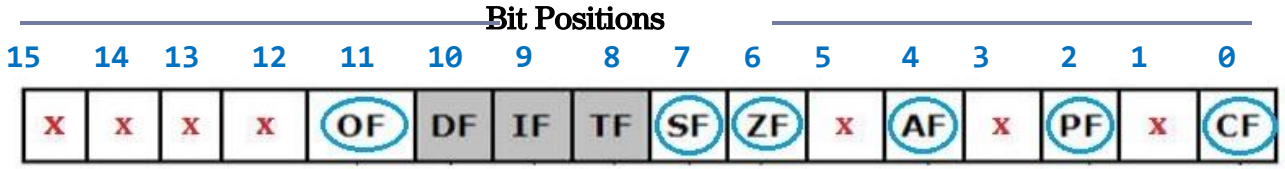
Dept. of Computer Science and Engineering
BRAC University
**CSE 341 Team**

# Lecture References:

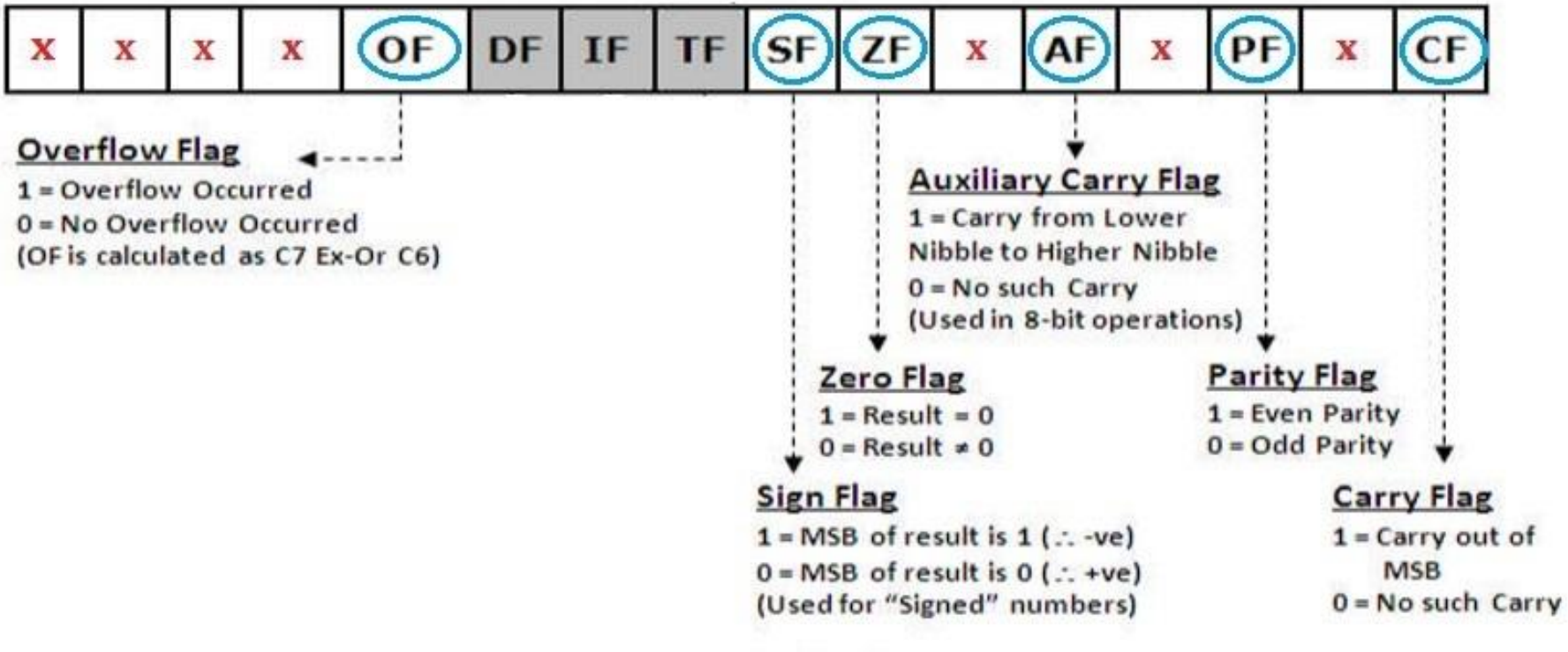
- **Book:**
  - *Microprocessors and Interfacing: Programming and Hardware, Chapter # 2,* **Author:** Douglas V. Hall

# 8086 Flag Register

- 16-**Bit register**
  - **7 bits are undefined/unused (marked by red x in the figure below)**
  - **6 status/condition flags (marked by blue circles)**
  - **3 control flags (those in grey boxes)**
- The condition flags are set (1) or reset (0) depending on the result of an arithmetic/logical operation.
- Control flags control the operations of the CPU



# Status Flags




Flags are useful in programming e.g. when writing conditions such as:

- If answer is zero, do ... else .... // zero flag comes in hand here
- If answer is less than zero, do ... else .... // sign flag can be used here

# Status Flags

- **Zero Flag (ZF) → 1 if the *answer* is exactly zero.**

  **0 if the *answer* is not exactly zero.**

- **Parity Flag (PF) → 1 if the count of 1s in the *answer* is EVEN**

  **0 if the count of 1s in the *answer* is ODD**

* NB: If there is no 1s (number of 1s is zero) in the ans, then parity flag is

- **Sign Flag (SF) → 1 if the msb in the *ans* is 1 (negative number)**

  **0 if the msb in the *ans* is 0 (positive number)**

# Be Careful

If we add two n bits numbers and we get an answer of (n+1) bits ,

We have to *consider the lowest n bits* in the answer.

**Table for consideration:**

| | 8 Bit Register | 16 bit Register |
|---|---|---|
| **SF** | **Lowest 8** | **Lowest 16** |
| **ZF** | **Lowest 8** | **Lowest 16** |
| **PF** | **Lowest 8** | **Lowest 8** |

# Status Flags

- **Auxiliary Flag (AF) → 1 if there is UNSIGNED overflow from lower nibble. Otherwise 0.**
- **Carry Flag (CF) -** is set to '1' when there is an unsigned overflow. E.g. when you add bytes 255 + 1 (result is not in range 0...255). When there is no overflow this flag is reset to 0.
- **Overflow Flag (OF) -** set to '1' when there is a signed overflow. For example, when you add bytes 100 + 50. (see next slide)

# Overflow Flag (OF)

1 → ONLY carry from 2nd last msb to last msb OR
ONLY carry from last msb to beyond msb (1 bit left of msb)

---

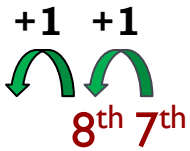
0 → Both carry from 2nd last msb to last msb AND carry from last msb to beyond msb. OR

No Carry from 2nd last msb to last msb AND also No carry from last msb to beyond msb

# Status Flags

- **Auxiliary Flag (AF) -** set to '1' when there is an unsigned overflow for low nibble (4 bits).
- **Carry Flag (CF) -** is set to '1' when there is an unsigned overflow. E.g. when you add bytes 255 + 1 (result is not in range 0...255). When there is no overflow this flag is reset to 0.
- **Overflow Flag (OF) -** set to '1' when there is a signed overflow. For example, when you add bytes 100 + 50. (see next slide)

# Overflow Flag (OF) Example




1 1 0 0 0 0 0 0

0 1 0 0 0 0 0 0

---

1 0 0 0 0 0 0 0

9th Bit

2nd Last MSB = 7th

Last MSB = 8th

Beyond MSB = 9th

CF = 1

OF = 0

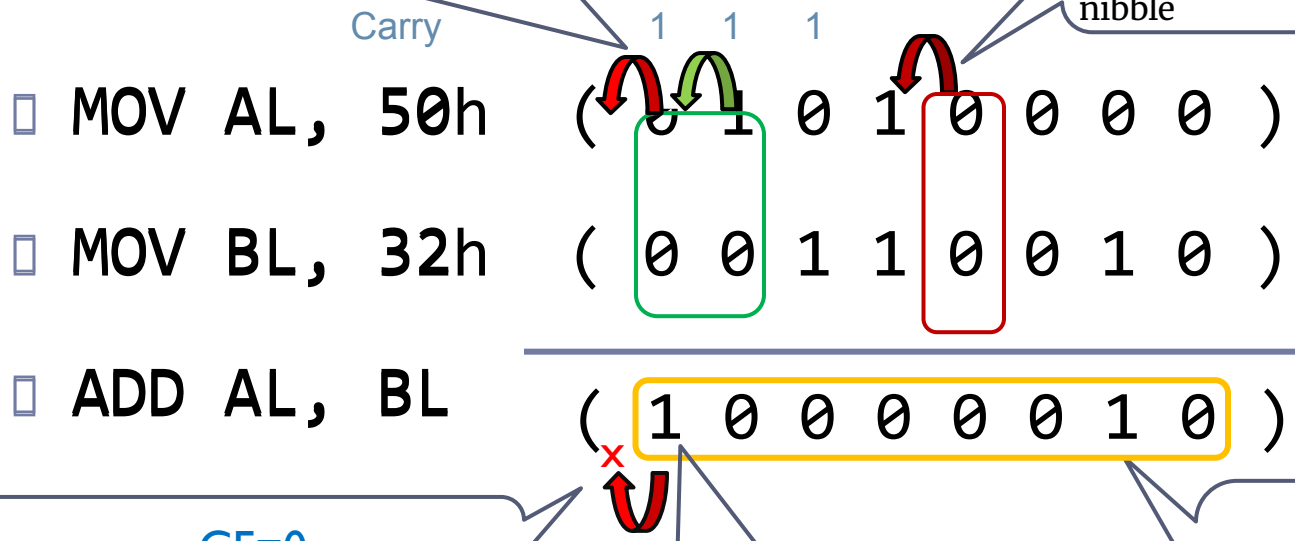
# Status Flags

**NB:**
OF is **set to 1** if there is a carry from:
- **the 7th bit to the 8th bit ONLY or**
- **from the 8th bit to the 9th bit ONLY**

If there is a carry from the **7th bit to the 8th** bit **and** from the **8th bit to the 9th bit** at **THE SAME TIME** then **OF = 0**

**OF=1**
Set because there is a carry from the 7th bit to the 8th bit ONLY

**AF=0**
Reset because there is NO carry from the lower nibble to the upper nibble




**CF=0**
Reset because the answer has NO carry

**SF=1**
Set because the MSB is 1 indicating a negative answer

**ZF=0**
Reset because the answer is NOT zero

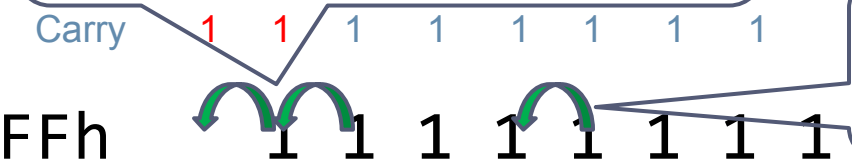
**PF=1**
Set because the answer has an EVEN number of 1s

# Status Flags

**NB:**
OF is **reset to 0** if:
- there is a **carry** from **the** the **7th bit to the 8th** bit **and** from the **8th bit to the 9th bit** at **THE SAME TIME** or
- there is **NO carry** from **the** the **7th bit to the 8th** bit **and** from the **8th bit to the 9th bit** at **THE SAME TIME**

**OF=0**
Reset because there is a carry from the **7th bit to the 8th** bit **and** from the **8th bit to the 9th bit** at **THE SAME TIME**

**AF=1**
Set because there IS a carry from the lower nibble to the upper nibble




**CF=1**
Set because the answer has a carry

**SF=0**
Reset because the MSB is 0 indicating a positive answer

**ZF=1**
Set because the answer IS zero

**PF=1**
Set because the answer has an EVEN number of 1s

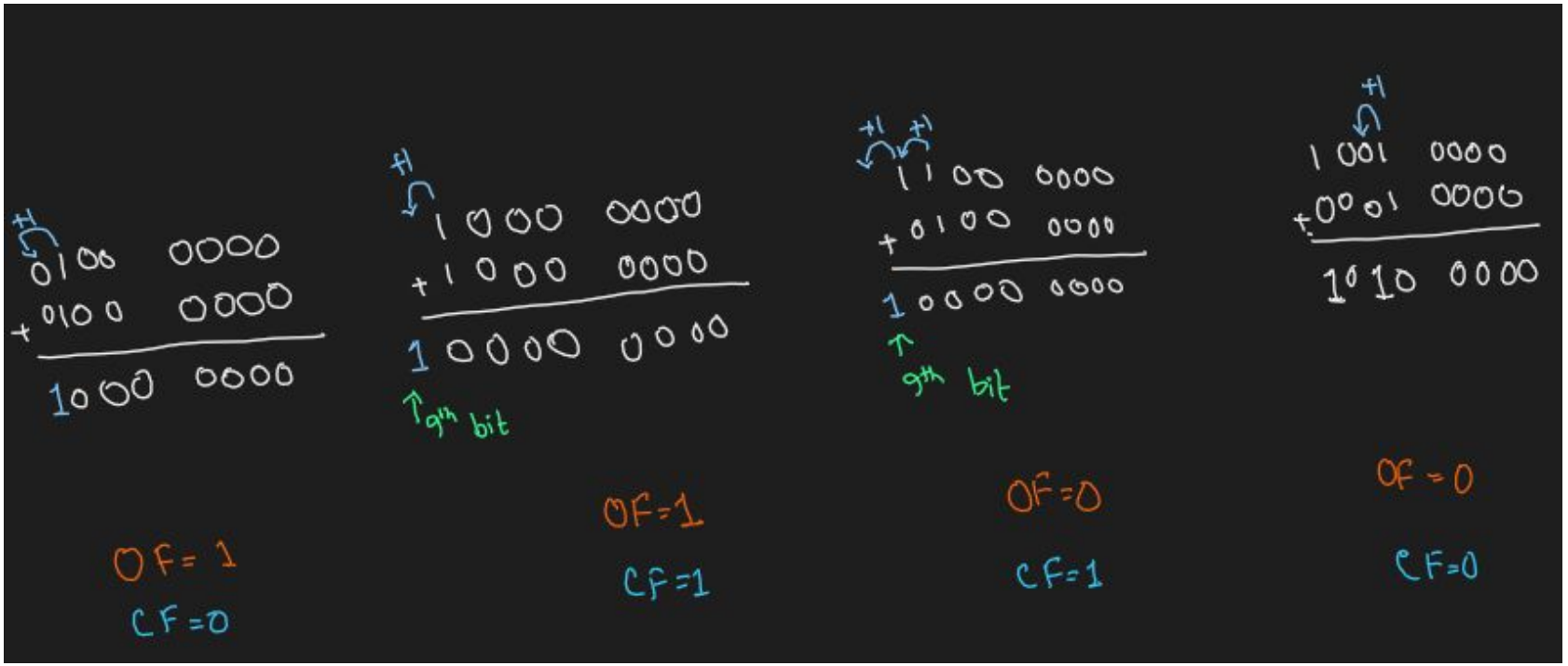
# Question

- Let's say we are adding two numbers of n digits. The resultant is (n+1). The carry flag is always 1.

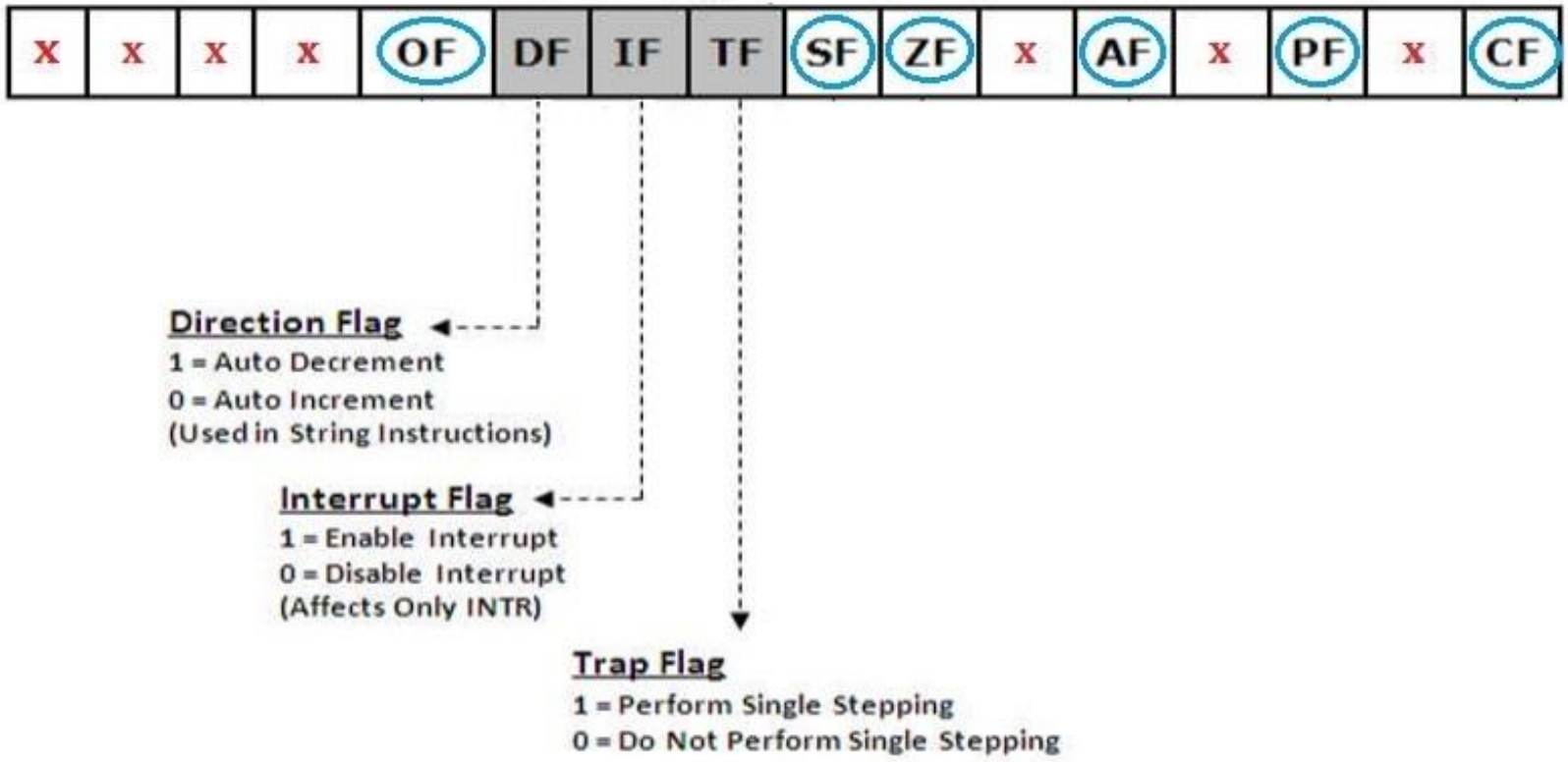
  (True / False)

- Will the carry flag and overflow flag value be always opposite. (Yes / No)

# OF and CF flags are INDEPENDENT



# Control Flags:



# Control Flags:

- **Trap Flag (TF)** - Used for on-chip single-step debugging.
- **Interrupt enable Flag (IF)** - when this flag is set to '1' CPU reacts to interrupts from external devices.
- **Direction Flag (DF)** - this flag is used by some instructions to process data chains, when this flag is set to '0' - the processing is done forward, when this flag is set to '1' the processing is done backward.

# Quiz: Status Flag Values?

- `MOV AX, ABCDh`
- `MOV BX, 9876h`
- `ADD AX, BX`

FFCDh

FFXYh

Given PF = 0 & AF = 1 , Maximum value for XY

Given PF = 0 & AF = 0 , Maximum value for XY

Given PF = 1 & AF = 0 , Maximum value for XY

Given PF = 1 & AF = 1 , Maximum value for XY

# Thank You !!